



①⑨ BUNDESREPUBLIK  
DEUTSCHLAND



DEUTSCHES  
PATENT- UND  
MARKENAMT

⑫ **Offenlegungsschrift**  
⑩ **DE 102 14 553 A 1**

⑤⑦ Int. Cl. 7:  
**G 06 F 9/45**

②① Aktenzeichen: 102 14 553.9  
②② Anmeldetag: 2. 4. 2002  
④③ Offenlegungstag: 16. 10. 2003

DE 102 14 553 A 1

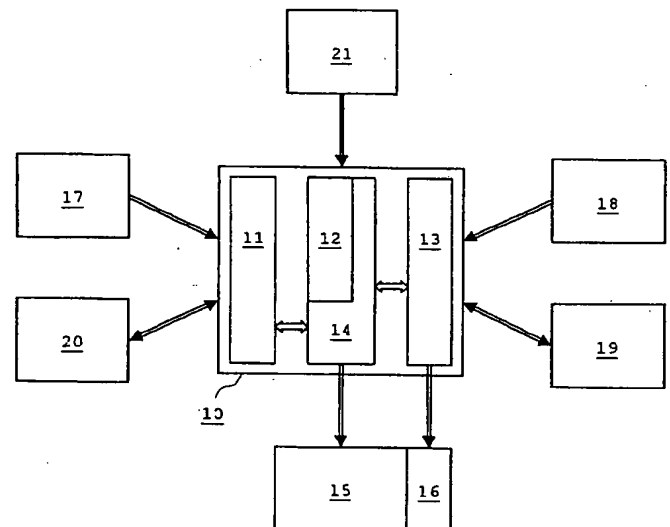
⑦① Anmelder:  
Siemens AG, 80333 München, DE

⑦② Erfinder:  
Höfler, Werner, 90542 Eckental, DE; Tiegelkamp,  
Michael, 90602 Pyrbaum, DE

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen**

⑤④ Einrichtung und Verfahren zur automatischen Generierung von Automatisierungssoftware

⑤⑦ Es wird eine Einrichtung und ein Verfahren zur automatischen Generierung von Automatisierungssoftware mit einem zumindest einen Softwaregenerator (11) und einen Projektgenerator (12) umfassenden Generator (10) angegeben, wobei mittels des Softwaregenerators (11) anhand von vorgebbaren oder vorgegebenen ersten Daten, wie insbesondere Planungsdaten (17), Programmwissen (18) und Verwaltungswissen (19) eine Entwicklungsumgebung (14) generierbar ist und unter Verwendung der Entwicklungsumgebung (14) mittels des Projektgenerators (12) anhand von vorgebbaren oder vorgegebenen zweiten Daten, wie insbesondere Planungsdaten (17), Erfahrungsdaten (20) und Hardwarewissen (21), eine Steuerungssoftware (15) generierbar ist.



DE 102 14 553 A 1

[0001] Die Erfindung bezieht sich auf eine Einrichtung zur automatischen Generierung von Automatisierungssoftware oder Teilen, wie insbesondere Strukturen und derselben. Sie bezieht sich weiter auf ein korrespondierendes Verfahren zur automatischen Generierung solcher Automatisierungssoftware.

[0002] In der älteren deutschen Patentanmeldung 102 06 691 wird ein Verfahren zur Nutzung zumindest eines in einem Speicher eines Rechners vorgehaltenen Dienstes durch einen Nutzer angegeben. Als Dienst wird dabei das Bereitstellen einer den Spezifikationen des Nutzers entsprechenden Software bezeichnet. Der Nutzer meldet sich bei einer einen Zugang zu dem Dienst überwachenden Verwaltungseinheit an und spezifiziert den gewünschten Dienst. Darauf ermittelt die Verwaltungseinheit zum Abruf eines der Spezifikation entsprechenden Dienstes dazu erforderliche Einzelkomponenten und generiert aus diesen die Software. Darüber hinaus generiert die Verwaltungseinheit für jede erforderliche Einzelkomponente einen Eintrag in einem dem Nutzer zugeordneten Lizenzdatensatz einer Lizenzdatenbank.

[0003] In der älteren deutschen Patentmeldung 102 09 640 wird ein Verfahren zur Generierung eines Automatisierungsprogramms aus einem HMI-Programm (HMI = "human machine interface") mittels eines Generators angegeben, wobei der Generator Strukturen des HMI-Programms erkennt und in Strukturen des Automatisierungsprogramms umsetzt.

[0004] Der Erfindung besteht in der Vereinfachung der Abläufe bei der Erstellung von Automatisierungssoftware.

[0005] Eine solche Vereinfachung der Abläufe bei der Erstellung von Automatisierungssoftware wird mit einer Einrichtung, wie sie in Anspruch 1 angegeben ist, erreicht. Dazu ist eine Einrichtung zur automatischen Generierung von Automatisierungssoftware mit einem zumindest einen Softwaregenerator und einen Projektgenerator umfassenden Generator vorgesehn, wobei mittels des Softwaregenerators anhand von vorgebbaren oder vorgegebenen ersten Daten, wie insbesondere Planungsdaten, Programmwissen und Verwaltungswissen eine Entwicklungsumgebung generierbar ist und wobei unter Verwendung der Entwicklungsumgebung mittels des Projektgenerators anhand von vorgebbaren oder vorgegebenen zweiten Daten, wie insbesondere Planungsdaten, Erfahrungsdaten und Hardwarewissen, eine Steuerungssoftware generierbar ist. Eine solche Vereinfachung wird gleichfalls mit einem korrespondierenden Verfahren, wie es in Anspruch 7 definiert ist, erreicht.

[0006] Die Erfindung geht von der Erkenntnis aus, dass sich Weiterentwicklungen an Software stets auf die Hinzufügung weiterer Leistungsmerkmale bezieht, mit denen der Anwender die eigentliche Aufgabe vermeintlich einfacher oder schneller lösen kann. Die Erfindung stellt jedoch die Aufgabe in den Vordergrund und sieht die automatische, zur Lösung der Aufgabe optimierte Generierung von Automatisierungssoftware vor. Dabei werden die derzeitigen Möglichkeiten beim Zugriff auf Daten verschiedenster Art ausgenutzt. Mit Hilfe leistungsfähiger Suchalgorithmen, wie sie zum Beispiel bei der Nutzung des Internets Verwendung finden, können umfangreiche Datenbestände in Bezug auf vorgegebene oder vorgebbare Schlüsseldaten, wie sie z. B. bei der Beschreibung der Aufgabe in Form von Schlüsselworten wie z. B. "Brauerei", "Stahlwalzwerk", etc. vorkommen, durchsucht werden. Daneben beobachtet man den Trend zur Verwendung standardisierter Schnittstellen. Diese erlauben z. B. eine Weiterverarbeitung von CAD-Daten in der Form, dass anhand der CAD-Daten und anhand von mit einzelnen

Datensätzen solcher CAD-Daten verknüpften Bibliotheksinformationen sämtliche in einem zu automatisierenden technischen Prozess enthaltenen Komponenten wie Motoren, Ventile, Endschalter, etc. ermittelbar sind. Anhand von vorgegebenen oder vorgebbaren Regeln ist jede Komponente z. B. der Kategorie der Aktoren oder der Kategorie der Sensoren zuordenbar. Für jeden Aktor sind eine geeignete Anzahl von Ausgängen einer Automatisierungshardware und für jeden Sensor entsprechend eine geeignete Anzahl von Eingängen vorzusehen, so dass damit bereits in groben Zügen das zu bearbeitende Datenvolumen feststeht und insbesondere ein zentrales Automatisierungsgerät nach Kriterien wie Rechenleistung und Speicherkapazität automatisch ausgewählt werden kann.

[0007] Demgegenüber wird Automatisierungssoftware heute mit käuflich erwerblichen, allgemein gebräuchlichen Standard-Entwicklungsumgebungen erstellt. Solche Standard-Entwicklungsumgebungen stellen gleichsam wie Konfektionsware eine Anzahl von Funktionen und Diensten zur Verfügung, ohne dabei individuelle Bedürfnisse einzelner Anwender berücksichtigen zu können. Gemäß der Erfindung ist zur automatischen Generierung von Automatisierungssoftware oder Teilen, wie insbesondere Strukturen, derselben zunächst eine Generierung einer individuellen Entwicklungsumgebung und nachfolgend unter Verwendung der individuell generierten Entwicklungsumgebung die Generierung einer Steuerungssoftware vorgesehen.

[0008] Bei der Steuerungssoftware handelt es sich um diejenige Automatisierungssoftware, die einen konkreten technischen Prozess, z. B. eine Fertigungsstraße, wie ein Walzgerüst oder eine Folge von Bestückungsautomaten, steuert und/oder überwacht und schließt ggf. eine graphische Benutzeroberfläche z. B. mittels eines Bedien- und Beobachtungssystems oder einer Mensch-Maschine-Schnittstelle (HMI) ein.

[0009] Bei der individuellen Entwicklungsumgebung handelt es sich um die Zusammenstellung sämtlicher zur Generierung der Steuerungssoftware notwendiger Komponenten, wie Bibliotheken oder Bibliotheksbestandteilen oder deren Äquivalente, Treiber zur Ansteuerung bestimmter Prozess- oder Automatisierungskomponenten, wie Netzwerkverbindungen, Kommunikationsschnittstellen, etc. sowie ein Transformationsprogramm, das nach Art eines Compilers, Interpreters oder dgl. zum Generieren der auf einer Zielhardware ausführbaren Steuerungssoftware vorgesehen ist.

[0010] Vorteilhafte Ausgestaltungen der Erfindung sind Gegenstand der Unteransprüche. Danach ist vorgesehen, dass der Generator weiter einen Anlagenkonfigurator umfasst und dass mittels des Anlagenkonfigurators anhand von vorgebbaren oder vorgegebenen dritten Daten, wie insbesondere Hardwarewissen, eine Konfiguration einer Automatisierungshardware ermittelbar ist. Damit wird die automatische Erstellung von Automatisierungssoftware abgerundet, indem zunächst die Entwicklungsumgebung und unter deren Verwendung die eigentliche Steuerungssoftware generiert wird, die mit Verfügbarkeit der Konfiguration der Automatisierungshardware direkt zum Ablauf gebracht werden kann.

[0011] Nachfolgend wird ein Ausführungsbeispiel der Erfindung anhand der Zeichnung näher erläutert. Einander entsprechende Gegenstände oder Elemente sind in allen Figuren mit den gleichen Bezugszeichen versehen.

[0012] Darin zeigen

[0013] Fig. 1 eine schematische Darstellung eines Generators zur automatischen Generierung von Automatisierungssoftware mit den dazu erforderlichen Daten,

[0014] Fig. 2 einzelne Verfahrensschritte bei der automatischen Generierung von Automatisierungssoftware und

[0015] Fig. 3 ein Diagramm des Datenflusses bei der auto-

matischen Generierung von Automatisierungssoftware.

[0016] Fig. 1 zeigt in einer schematischen Übersicht einen Generator 10 mit einem Softwaregenerator 11, einen Projektgenerator 12 und einem Anlagenkonfigurator 13.

[0017] Der Softwaregenerator 11 ist zur Generierung einer Entwicklungsumgebung 14 und der Projektgenerator 12 zur Generierung einer Steuerungssoftware 15 vorgesehen. Der Anlagenkonfigurator 13 ist zur Ermittlung einer geeigneten Konfiguration 16 zum Betrieb der Steuerungssoftware 15 auf einer Automatisierungshardware vorgesehen. Bei der nicht dargestellten Automatisierungshardware handelt es sich um einzelne Automatisierungsgeräte, wie z. B. eine speicherprogrammierbare Steuerung oder einen Prozessrechner oder einen Verbund solcher Geräte, mit z. B. dezentralen Peripherieanschlüssen und/oder einem Leitrechner mit z. B. Produktplanungsfunktionalitäten. Ggf. nimmt der Anlagenkonfigurator 13 das Konfigurieren der Automatisierungshardware auch selbst vor, indem er die jeweilige Konfiguration 16, das heißt die Konfigurationsdaten, an die Automatisierungshardware übermittelt. Die Steuerungssoftware 15 steuert die Automatisierungshardware und steuert und/oder überwacht damit einen technischen Prozess (nicht dargestellt), wie z. B. eine Schweißstraße (Automobilproduktion oder dgl.) oder ein Walzgerüst (Stahlerzeugung).

[0018] Die Generierung der Entwicklungsumgebung 14 durch den Softwaregenerator 11 erfolgt anhand von vorgegebenen oder vorgegebenen ersten Daten, wie insbesondere Planungsdaten 17, Programmwissen 18 und Verwaltungswissen 19. Die Planungsdaten 17 umfassen Daten wie Konstruktionsdaten, wie sie z. B. von einem CAD-System verarbeitet oder bereit gestellt werden. Daneben oder zusätzlich umfassen die Planungsdaten 17 eine Aufgabenbeschreibung und/oder eine Problembeschreibung und/oder Daten zu vorhergehenden Projekten. Das Programmwissen 18 umfasst Daten zu Funktionen wie z. B. Leistungsumfang oder Verfügbarkeit, Strukturen, Versionen und/oder Abhängigkeiten, z. B. einzelner Funktionen untereinander. Das Verwaltungswissen 19 umfasst Daten für eine Nutzungsverwaltung, Daten für eine Abrechnung und/oder Daten für einen Aktualisierungsservice (Update).

[0019] Die jeweils zur Verfügung stehenden oder zur Verfügung gestellten ersten Daten werden mittels des Softwaregenerators 11 analysiert. Anhand z. B. der Konstruktionsdaten ist ermittelbar, dass im technischen Prozess z. B. frequenzgeregelte Antriebe vorhanden sind. Solche Antriebe erfordern spezielle Aktoren. Diese wiederum erfordern spezielle Software zu deren Ansteuerung. Die damit als erforderlich ermittelte Software ist eine notwendige Komponente der automatisch zu erstellenden Entwicklungsumgebung 14. Anhand der Aufgaben- oder Problembeschreibung, insbesondere in Verbindung mit Daten zu vorhergehenden Projekten, ist anhand von Teilaufgaben oder Teilproblemen, ggf. unter Berücksichtigung der jeweiligen Behandlung bei vorangehenden Projekten, ermittelbar, welche weiteren Komponenten für die Entwicklungsumgebung 14 notwendig sind. Für jede für die Entwicklungsumgebung 14 notwendige Komponente ist anhand des Programmwissens 18 ermittelbar, welche Komponenten, wie Softwaremodule, Bibliotheken oder dgl., für die Entwicklungsumgebung 14 erforderlich sind. Zu diesem Zweck wird z. B. auf Daten zum Funktionsumfang oder auf Daten zu Strukturen zurückgegriffen. Häufig macht eine für die Entwicklungsumgebung 14 erforderliche Komponente die Aufnahme weiterer Komponenten erforderlich, weil Abhängigkeiten, wie Importbeziehungen, zwischen der erforderlichen Komponente und der oder jeder weiteren Komponente bestehen. Zu diesem Zweck wird z. B. auf Daten zu Abhängigkeiten oder auf Daten zu Strukturen zurückgegriffen. Des Weiteren muss

die Kompatibilität der in die Entwicklungsumgebung 14 eingehenden Komponenten gewährleistet sein. Dazu wird z. B. auf Daten zu Versionen und/oder Abhängigkeiten zurückgegriffen. Schließlich muss die Verfügbarkeit und ggf. die Aktualität der einzelnen erforderlichen Komponenten gewährleistet sein. Dazu ist ein Zugriff auf Daten zur Nutzungsverwaltung und/oder Daten für einen Aktualisierungsservice vorgesehen. Um die verwendeten Komponenten abrechnen oder deren Verwendung berechnen zu können ist ein Zugriff auf Abrechnungsdaten vorgesehen. Nach Berücksichtigung sämtlicher zur Verfügung stehender erster Daten steht schließlich eine individuell an die jeweilige Aufgabe oder Problemstellung angepasste Entwicklungsumgebung 14 zur Verfügung.

[0020] Die Generierung der Steuerungssoftware 15 durch den Projektgenerator 12 erfolgt unter Verwendung der Entwicklungsumgebung 14 anhand von vorgegebenen oder vorgegebenen zweiten Daten, wie insbesondere die Planungsdaten 17 sowie Erfahrungsdaten 20 und Hardwarewissen 21. Anhand der Planungsdaten 17, wie insbesondere Aufgaben- und Problembeschreibung, ggf. in Verbindung mit Daten zu vorhergehenden Projekten, lässt sich eine Struktur der Steuerungssoftware 15 im Hinblick auf erforderliche Funktionen und eine Reihenfolge zum Aufruf solcher Funktionen ermitteln. Je detaillierter z. B. die Daten zu vorhergehenden Projekten sind, desto umfassender können die ermittelten Strukturen mit Programmanweisungen gefüllt werden. Wenn eine automatische Generierung solcher Programmanweisungen nicht möglich ist, kann vorgesehen sein, dass die Strukturen zumindest mit Informationen, beispielsweise in Form von Kommentarzeilen, gefüllt werden, die einen Programmierer bei einer späteren Überarbeitung darüber informieren, welche Operationen an einer konkreten Stelle der Steuerungssoftware 15 vorzusehen sind.

[0021] Die Erfahrungsdaten 20 umfassen zum Beispiel Daten zu Experten- oder Branchenwissen. Solche Daten können auch aktuelle Trends berücksichtigen. Anhand dieser Daten kann zum Beispiel automatisch ermittelt werden, dass in kritischen Industrieumgebungen explosionsgeschützte Feldgeräte oder Feldgeräte mit einer besonderen Schutzart, z. B. IP65, vorzusehen sind. Solche Feldgeräte sind für die jeweiligen Anbieter im Vergleich zu Standard-Feldgeräten Sonderlösungen. Wenn Standard-Feldgeräte z. B. mit einer Kanalgranularität von 1 oder 2 Bit (Digital I/O) erhältlich sind, sind spezielle Feldgeräte nur in wenigen Konfigurationen, z. B. 8-Bit und 16-Bit (Digital I/O), verfügbar. Das heißt unter Berücksichtigung von Branchen- oder Expertenwissen kann sich eine andere Konfiguration der Feldgeräte ergeben, als dies ohne Berücksichtigung solcher Informationen der Fall wäre. Wenn ein oder mehrere spezielle Feldgeräte mit acht digitalen Ein-/Ausgabekanälen als erforderlich ermittelt werden, erfordert dies auch entsprechende Anpassungen für die diese Feldgeräte ansteuernde Steuerungssoftware 15.

[0022] Des Weiteren oder alternativ können die Erfahrungsdaten 20 auch eine Herstellerbibliothek mit z. B. Musterlösungen umfassen. Anhand solcher Musterlösungen können automatisch generierte Strukturen der Steuerungssoftware 15 mit geeigneten Programmanweisungen gefüllt werden, um z. B. die korrekte Bedienung eines konkreten Feldgerätes zu gewährleisten.

[0023] Schließlich können die Erfahrungsdaten 20 auch eine Erfahrungsdatenbank umfassen. Aus einer solchen Erfahrungsdatenbank können Informationen darüber entnommen werden, ob sich eine konkrete, ggf. auch automatisch generierte Automatisierungssoftware bei einem vergleichbaren Problem oder Teilproblem oder einer vergleichbaren Aufgabe oder Teilaufgabe als geeignet herausgestellt hat.

Damit werden nach Art einer evolutionären Strategie geeignete Lösungen von ungeeigneten oder weniger geeigneten unterscheidbar. Für die jeweilige Steuerungssoftware 16 werden vorzugsweise als geeignet erkannte Lösungen auf deren Verwendbarkeit überprüft.

[0024] Die Erfahrungsdaten 20 werden kontinuierlich aktualisiert, denn jede generierte Steuerungssoftware 16 liefert neue Erfahrungsdaten 20, die für nachfolgende Projekte nützlich sein können.

[0025] Einzelne Programmanweisungen in der Steuerungssoftware 15 beziehen sich auf eine Ansteuerung der Automatisierungshardware. Um eine korrekte Ansteuerung der Automatisierungshardware oder einzelner Automatisierungskomponenten der Automatisierungshardware, wie z. B. Ein-/Ausgabebaugruppen zum Aufnehmen von Daten aus dem technischen Prozess oder zum Übermitteln von Steuerinformationen an den technischen Prozess, zu gewährleisten, umfassen die zweiten Daten auch Hardwarewissen 21. Anhand des Hardwarewissens 21 ist z. B. ermittelbar, welche Maßnahmen zum Aktivieren eines bestimmten Ausgangs eines bestimmten Feldgerätes der Automatisierungshardware erforderlich sind. Damit lassen sich automatisch die entsprechenden Programmanweisungen für die Steuerungssoftware 15 generieren um z. B. den dritten Ausgang eines Feldgerätes zu aktivieren. Ein Feldgerät ist üblicherweise über einen Feldbus mit einem zentralen Automatisierungsgerät, wie einer speicherprogrammierbaren Steuerung, verbunden. Anhand des Hardwarewissens lassen sich automatisch für die Steuerungssoftware 15 auch diejenigen Programmanweisungen generieren, um das Übertragungsprotokoll für den jeweils eingesetzten Feldbus zu bedienen.

[0026] Die automatische Generierung der Konfiguration 16 durch den Anlagenkonfigurator 13 erfolgt unter Verwendung dritter Daten, wie z. B. Hardwarewissen 21. Bei der Generierung der Steuerungssoftware 16 ergibt sich Art und Umfang der benötigten Automatisierungshardware. Danach ist z. B. bekannt, über welche Leistungsfähigkeit, z. B. im Hinblick auf Verarbeitungsgeschwindigkeit oder Speicherkapazität, ein zentrales Automatisierungsgerät verfügen muss. Damit ist anhand von zum Hardwarewissen 21 gehörenden, ggf. auch herstellerübergreifenden Katalogdaten ermittelbar, welches Automatisierungsgerät diese Anforderungen erfüllt. Nachdem auf diese Weise das konkrete Automatisierungsgerät ermittelt ist, ist anhand von weiterem Hardwarewissen 21 auch ermittelbar, wie und in welchem Umfang das Automatisierungsgerät konfigurierbar ist. Anhand der Erfordernisse der Steuerungssoftware 15 lässt sich die jeweils erforderliche Konfiguration 16 ermitteln und ggf. auch automatisch an das Automatisierungsgerät in passender Form übermitteln. Ähnlich ergibt sich aus Art und Umfang der benötigten Automatisierungshardware z. B. auch, ob und in welcher Form die einzelnen Automatisierungskomponenten untereinander durch einen Feldbus verbunden werden müssen. Aufgrund des anhand der Steuerungssoftware 15 ermittelbaren zu erwartenden Datenvolumens ergibt sich eine Datenübertragungsrate für den Feldbus. Die Datenübertragungsrate beeinflusst ggf. das verwendbare Protokoll. Die auf diese Weise ermittelten Konfigurationsdaten im Hinblick auf Protokoll und Datenübertragungsrate gehen ebenfalls in die Konfiguration 16 ein. Eine Übermittlung der Konfiguration 16 an die jeweiligen Komponenten der Automatisierungshardware umfasst dann auch das Konfigurieren jedes an den Feldbus angeschlossenen Automatisierungsgerätes.

[0027] Nach all diesen Schritten steht eine automatisch generierte Automatisierungssoftware zur Verfügung, die die Entwicklungsumgebung 14, die Steuerungssoftware 15 und ggf. die Konfiguration 16 umfasst. Die ersten Daten beein-

flussen dabei die automatische Generierung der Entwicklungsumgebung 14. Die zweiten Daten und die Entwicklungsumgebung 14 beeinflussen die automatische Generierung der Steuerungssoftware 15. Die dritten Daten und die Steuerungssoftware 15 beeinflussen die automatische Generierung der Konfiguration 16. Damit wirkt sich also jede Änderung an den ersten Daten auf die Entwicklungsumgebung 14 und somit zumindest mittelbar auf die Steuerungssoftware 15 und die Konfiguration 16 aus. Entsprechend wirkt sich jede Änderung an den zweiten Daten auf die Steuerungssoftware 15 und zumindest mittelbar auf die Konfiguration 16 aus. Schließlich wirkt sich jede Änderung an den dritten Daten auf die Konfiguration 16 aus. Als Änderung der ersten, zweiten oder dritten Daten ist z. B. eine Änderung des Umfangs und/oder der Verfügbarkeit der jeweiligen Daten aber auch Aktualisierungen der jeweiligen Daten zu verstehen. Solche Änderungen haben selbstverständlich nur dann Auswirkungen auf die automatisch generierte Automatisierungssoftware, wenn die Änderungen sich auf solche Daten beziehen, die bei der automatischen Generierung verwendet wurden oder verwendbar sind.

[0028] Es ergibt sich insgesamt ein iterativer Prozess bei der automatischen Generierung von Automatisierungssoftware, der anhand von Fig. 2 veranschaulicht wird. Ein erster Verfahrensschritt 22 umfasst das Festlegen der verwendeten oder verwendbaren ersten, zweiten und dritten Daten. Dazu gehört eine manuelle Spezifikation von z. B. Planungsdaten 17 mit einer Beschreibung des zu automatisierenden technischen Prozesses. Ferner oder alternativ gehört dazu ein Import importierbarer Daten oder das Bereitstellen einer Möglichkeit zum Import solcher Daten. Schließlich kann der erste Verfahrensschritt 22 auch das Auswählen oder Festlegen einer Importschnittstelle umfassen, mittels derer erste, zweite oder dritte Daten importiert werden können.

[0029] Ein zweiter Verfahrensschritt 23 umfasst das automatische Generieren der Entwicklungsumgebung 14, der Steuerungssoftware 15 und ggf. der Konfiguration 16. Ein dritter Verfahrensschritt 24 umfasst das Bearbeiten und Erweitern der Steuerungssoftware 15 mittels der Entwicklungsumgebung 14. In einem vierten Verfahrensschritt 25 wird die generierte Automatisierungssoftware, also die Steuerungssoftware 15 und ggf. die Konfiguration 16 zur Automatisierungshardware übertragen.

[0030] Ein Anwender kann in jedem Verfahrensschritt 22-24 Änderungen vornehmen und darauf die automatische Generierung der Automatisierungssoftware erneut veranlassen. Der Anwender führt die einzelnen Verfahrensschritte sowie darin erforderliche Anpassungen oder Ergänzungen so oft durch, bis schließlich die fertige Automatisierungssoftware zur Verfügung steht. Je umfassender die ersten, zweiten und dritten Daten sind, desto detaillierter kann die Automatisierungssoftware automatisch durch den Generator 10 generiert werden. Nicht automatisch generierbare Komponenten oder Sequenzen der Automatisierungssoftware ergänzt der Anwender im dritten Verfahrensschritt 24. Dabei wird er nach Möglichkeit durch automatisch in die Steuerungssoftware 15 anstelle von nicht automatisch generierbaren Sequenzen eingefügte Kommentare unterstützt. Diese Kommentare enthalten Hinweise auf die Funktionalität der nicht automatisch generierbaren Sequenz und vorzugsweise Hinweise auf die dafür zu verwendenden Daten.

[0031] Der Generator 10 erzeugt eine individuell abgestimmte Automatisierungssoftware. Bei z. B. neuen Funktionen wird die erzeugte Automatisierungssoftware automatisch aktualisiert. Das heißt in der Entwicklungsumgebung 14 ist diese neue Funktion verfügbar, so dass sie ggf. auch Bestandteil der schließlich generierten Steuerungssoftware 15 wird. Für den Anwender entfallen damit Administrati-

onsaufgaben für Software, Projekte und Bibliotheken. Als Zusatzinformation wird festgehalten, welche Daten, das heißt z. B. welche Funktionen, für die automatische Generierung der Automatisierungssoftware tatsächlich benötigt wurden. Damit können etwaige Lizenzgebühren konkret anhand der jeweils tatsächlich in Anspruch genommenen Leistung berechnet werden. Der Anbieter von Automatisierungssoftware, etwa ein Anbieter von Entwicklungsumgebungen zur Erstellung von Automatisierungssoftware, vermeidet die Notwendigkeit, diverse Softwarepakete mit jeweils unterschiedlichem Leistungs- und Funktionsumfang in noch dazu häufig verschiedenen Versionen vorhalten und unterstützen zu müssen, da die Automatisierungssoftware und damit auch die jeweilige Entwicklungsumgebung individuell entsprechend der konkreten Anforderungen des Anwenders generiert werden.

[0032] Fig. 3 zeigt ein Diagramm des Datenflusses bei der automatischen Generierung von Automatisierungssoftware. Ähnlich wie in Fig. 1 ist der Generator 10 mit Softwaregenerator 11, Projektgenerator 12 und Anlagenkonfigurator 13 im Zentrum dargestellt. Der Generator 10 verarbeitet Eingabedaten 26 und erzeugt Ausgabedaten 27. Zu den Eingabedaten 27 gehören die ersten, zweiten und dritten Daten. Zu den ersten Daten gehören wiederum die Planungsdaten 17, das Programmwissen 18 und Verwaltungswissen 19. Zu den zweiten Daten gehören die Planungsdaten 17, Erfahrungsdaten 20 und Hardwarewissen 21. Zu den dritten Daten gehört zumindest das Hardwarewissen 21.

[0033] Anhand vorgegebener oder vorgegebener erster Daten generiert der Softwaregenerator 11 die Entwicklungsumgebung 14. Anhand vorgegebener oder vorgegebener zweiter Daten und unter Verwendung der Entwicklungsumgebung 14 generiert der Projektgenerators 12 die Steuerungssoftware 15. Der Anlagenkonfigurator 13 generiert schließlich anhand vorgegebener oder vorgegebener dritter Daten die Konfiguration 16 der jeweiligen Automatisierungshardware.

[0034] Die wesentliche Einflussmöglichkeit des Anwenders besteht einerseits in der Bereitstellung und Vervollständigung geeigneter erster, zweiter und dritter Daten und andererseits in Form einer Benutzereingabe 28, mit der der Anwender in jedem Verfahrensschritt 22-25 (Fig. 2) die automatische Generierung durch manuelle Ergänzungen oder Modifikationen beeinflussen kann.

[0035] Zur Verarbeitung der Eingabedaten 26 ist für den Generator 10 eine Vorverarbeitung 29 vorgesehen. Diese gewährleistet die Möglichkeit des Imports der ersten, zweiten oder dritten Daten indem entweder eine definierte Schnittstelle zum Zugriff auf diese Daten bedient wird oder indem die Daten in ein zur weiteren Verarbeitung z. B. durch den Softwaregenerator 11 oder den Projektgenerator 12 geeignetes Format konvertiert werden.

[0036] Des Weiteren berücksichtigt die Vorverarbeitung 29 auch vorgegebene oder vorgebbare und damit insbesondere veränderbare und erweiterbare Regeln einer nicht separat dargestellten Regeldatenbank. Mit Hilfe dieser Regeln, die je nach Bedarf automatisch ausgewählt und angewendet werden, ist eine Analyse und Auswertung der ersten, zweiten und dritten Daten möglich.

[0037] Eine dieser Regeln lautet z. B.: "Wenn dezentrale Feldgeräte vorgesehen oder erforderlich sind, ist eine Feldbusverbindung zwischen dem oder jedem dezentralen Feldgerät und einem zentralen Automatisierungsgerät vorzusehen." Weitere Regeln beziehen sich dann z. B. auf die Art der Feldbusverbindung z. B. im Hinblick auf den maximal erreichbaren Datendurchsatz. Andere Regeln können z. B. die Verwendung sicherheitsgerichteter Automatisierungskomponenten beeinflussen. Eine solche Regel könnte z. B.

lauten: "Bei kontinuierlichen, insbesondere nicht unterbrechbaren Prozessen sind sicherheitsgerichtete oder fehler-tolerante Automatisierungskomponenten vorzusehen". Eine solche Regel ist ein Beispiel für eine erweiterbare Regel, bei der der Anwender z. B. angeben kann, dass "ein chemischer Prozess zum Raffinieren von Rohöl ein solcher kontinuierlicher, nicht unterbrechbarer Prozess ist". Ob es sich beim jeweiligen technischen Prozess um einen zum Raffinieren von Rohöl handelt, ergibt sich anhand der ersten Daten, insbesondere anhand der Planungsdaten, z. B. anhand der Aufgaben- oder Problembeschreibung.

[0038] Die automatische Verarbeitung der ersten, zweiten und dritten Daten, das Ermitteln untereinander bestehender Beziehungen, die Auswahl passender Regeln etc. wird erleichtert, wenn sämtliche Daten mit normierten Kennungen versehen und daran identifizierbar sind. Das Format solcher Kennungen folgt vorzugsweise bestehenden oder sich abzeichnenden Standards. So hat z. B. das W3C (WWW-Consortium; WWW = World Wide Web) WebServices, also im Internet angebotene oder abrufbare Dienstleistungen, bezüglich des Nachrichtenformats und -austauschs, im Hinblick auf Directory-Services zur Speicherung von Dienstbeschreibungen (UDDI) sowie einer Sprache zur Dienstbeschreibung (WSDL) spezifiziert.

[0039] Solche Spezifikationen können auch für zur Identifikation und Zuordenbarkeit der ersten, zweiten und dritten Daten verwendet werden.

[0040] So können z. B. innerhalb der Planungsdaten 17 Daten zu vorhergehenden Projekten Kennungen zugeordnet werden, die Aufschluss darüber geben, mit welchen Softwarefunktionen einzelne Komponenten oder Sequenzen projektiert wurden. Kommt nun eine spätere Wiederverwendung solcher Komponenten oder Sequenzen für einen neuen Steuerungssoftware 15 in Betracht, lässt sich anhand der Kennungen automatisch ermitteln, welche Funktionalität z. B. in der Entwicklungsumgebung 14 zur Handhabung der übernehmbaren Komponenten oder Sequenzen vorhanden sein muss.

[0041] Ein weiteres Beispiel für die Verwendung standardisierter Kennungen zu Identifizierung und Zuordnung von Daten und Informationen zur automatischen Generierung von Automatisierungssoftware bezieht sich auf Planungsdaten 17 und dort genauer auf Konstruktionsdaten, wie sie beispielsweise von einem CAD-Programm geliefert werden und Hardwarewissen 21, wie z. B. Katalogdaten. Wenn Konstruktionsdaten mit eindeutigen, informativen Kennungen z. B. zur Anwendung und Funktionalität einzelner Komponenten, wie z. B. Motoren, Ventilen, Sensoren wie Tastern oder Lichtschranken, etc. versehen sind, können Katalogdaten oder auch vorangehende Projekte nach korrespondierenden Eigenschaften durchsucht werden und darin eine zielgerichtete Auswahl getroffen werden.

[0042] Kennungen der oben genannten Art können sich z. B. auf

- Branche, Anwendungsgebiet, Produkt,
- Fertigungsprinzip (Förderband, Rohrleitungen, ...)
- Projektstrukturierung (technologisch, Funktional, CPU orientiert, ...)
- Programmiersprache bzw. Programmiermethode (ablauforientiert, continuous, Kontaktplan, Funktionsplan)
- Regelalgorithmus, Regelparameter,
- Teilprozesse (Förderbandsteuerung, rühren, heizen, pressen, biegen, etc.)
- etc.

beziehen.

[0043] Alle benötigten Daten werden dem Generator 10 über Standardschnittstellen und Standardnetze zur Verfügung gestellt.

[0044] Als Beispiel dafür wird auf die erweiterbare Deklarationssprache ebXML verwiesen. ebXML ist zur Abwicklung von Geschäftsprozessen im das Internet vorgesehen. Es existieren Spezifikationen zu Nachrichtenformaten und -austausch, einem Registry- und Repositorydienst sowie Schablonen für bestimmte geschäftsspezifische Objekte und Abläufe. Die Basis des Nachrichtentransports bei WebServices wie auch mittlerweile bei ebXML ist das Simple Object Access Protocol (SOAP), ein XML-basiertes Protokoll zur RPC- bzw. nachrichtenorientierten Kommunikation, das typischerweise in HTTP gekapselt ist und einen zuverlässigen Nachrichtenaustausch gewährleistet.

[0045] Das W3C hat Web Services bezüglich des Nachrichtenformats und -austauschs, eines Directory-Services zur Speicherung von Dienstbeschreibungen (UDDI) sowie einer Sprache zur Dienstbeschreibung (WSDL) spezifiziert.

[0046] Nach einer Identifikation der für die Entwicklungsumgebung 14 und/oder für die Steuerungssoftware 15 als notwendig ermittelten Komponenten werden diese in den ersten, zweiten und dritten Daten gesucht und geeignete Komponenten zur Verwendung ausgewählt. Voraussetzung für die Verwendbarkeit ist eine feinmodulare Struktur der Software und der einzelnen Komponenten, der über vordefinierte Schnittstellen deren Kombination erlaubt.

[0047] Zur weiteren Generierung der Steuerungssoftware 15 wird in erster Linie auf vorhergehende Projekte innerhalb der Planungsdaten 17 zurückgegriffen. Die Auswahl unter einer Vielzahl vorangehender Projekte erfolgt im Hinblick auf eine möglichst weitgehende Übereinstimmung mit den jeweiligen konkreten Anforderungen. Aus vorangehenden Projekten übernehmbare Komponenten oder Sequenzen bilden eine Basis für die zu erstellende Automatisierungssoftware. Diese Basis wird mit als geeignet ermittelten Komponenten aus anderen zurückliegenden Projekten, Herstellerbibliotheken, Katalogen, etc. ergänzt.

[0048] Die Steuerungssoftware 15 kann aus einzelnen, unabhängig voneinander generierbaren und ausführbaren Programmen bestehen. Zur gemeinsamen Steuerung eines technischen Prozesses ist regelmäßig eine Kommunikation zwischen solchen grundsätzlich unabhängigen Programmen erforderlich. Die möglichen Kommunikationswege zwischen einzelnen Programmen wird über die Planungsdaten 17 ermittelt und über Standardmechanismen etabliert (net, Ethernet, ...)

[0049] Der Generator 10 umfasst schließlich noch eine Protokolleinheit 30 zum Aufzeichnen der verwendeten Daten. Die aufgezeichneten Daten umfassen auch Daten zu der oder jeder verwendeten Softwarekomponente und ob diese Softwarekomponente bei der Generierung der Entwicklungsumgebung 14 oder bei der Generierung der Steuerungssoftware 15 verwendet wurde. Anhand dieser Daten kann nach Zugriff auf ein Verwaltungssystem 31, das z. B. auf einem zentralen, insbesondere über das Internet erreichbaren Server vorgehalten ist, eine Rechnung 32 für den Anwender unter Berücksichtigung der verwendeten Daten erstellt werden.

[0050] Damit lässt sich die Erfindung kurz wie folgt darstellen:

Die Erfindung löst sich von starren Programmen und Bibliotheken und anschließender Eingabe von Daten zur Problemlösung. Statt dessen werden geeignete Daten oder ein Zugriff auf solche Daten bereit gestellt, mit denen eine automatische Generierung zunächst einer individuellen, auf das konkrete Automatisierungsproblem abgestellten Entwicklungsumgebung 14 und darauf unter deren Verwendung die

automatische Generierung der eigentlichen Steuerungssoftware 15 erfolgt. Dabei werden die Möglichkeiten des Datenzugriffs über Internet und standardisierte Programm- oder Datenschnittstellen, wie z. B. XML, auf bisher noch lokal gehaltene Informationen von Planungstools oder dgl. und Bibliotheken oder bestehenden Projekten ausgenutzt. Die Rahmenbedingungen des jeweiligen Automatisierungsvorhabens bilden dabei den Ausgangspunkt für die einzelnen, automatisch zu generierenden Elemente der Automatisierungssoftware, wie das eigentliche Steuerprogramm 15, ggf. ein ergänzendes HMI-Programm und weiterhin ggf. eine Anlagenkonfiguration 16.

#### Patentansprüche

1. Einrichtung zur automatischen Generierung von Automatisierungssoftware mit einem zumindest einen Softwaregenerator (11) und einen Projektgenerator (12) umfassenden Generator (10), wobei mittels des Softwaregenerators (11) anhand von vorgebbaren oder vorgegebenen ersten Daten, wie insbesondere Planungsdaten (17), Programmwissen (18) und Verwaltungswissen (19) eine Entwicklungsumgebung (14) generierbar ist und wobei unter Verwendung der Entwicklungsumgebung (14) mittels des Projektgenerators (12) anhand von vorgebbaren oder vorgegebenen zweiten Daten, wie insbesondere Planungsdaten (17), Erfahrungsdaten (20) und Hardwarewissen (21), eine Steuerungssoftware (15) generierbar ist.
2. Einrichtung nach Anspruch 1, wobei der Generator (10) weiter einen Anlagenkonfigurator (13) umfasst und wobei mittels des Anlagenkonfigurators (13) anhand von vorgebbaren oder vorgegebenen dritten Daten, wie insbesondere Hardwarewissen (21), eine Konfiguration (16) einer Automatisierungshardware ermittelbar ist.
3. Einrichtung nach Anspruch 1 oder 2, wobei die ersten Daten Planungsdaten (17), Programmwissen (18) und/oder Verwaltungswissen (19) umfassen.
4. Einrichtung nach Anspruch 1 oder 2, wobei die zweiten Daten Planungsdaten (17), Erfahrungsdaten (20) und/oder Hardwarewissen (21) umfassen.
5. Einrichtung nach Anspruch 1 oder 2, wobei die dritten Daten Hardwarewissen (21) umfassen.
6. Einrichtung nach einem der vorangehenden Ansprüche, wobei die Steuerungssoftware (16) eine Mensch-Maschine-Schnittstelle umfasst.
7. Verfahren zur automatischen Generierung von Automatisierungssoftware unter Verwendung eines zumindest einen Softwaregenerator (11) und einen Projektgenerator (12) umfassenden Generators (10), wobei der Softwaregenerator (11) anhand von vorgebbaren oder vorgegebenen ersten Daten, wie insbesondere Planungsdaten (17), Programmwissen (18) und Verwaltungswissen (19) eine Entwicklungsumgebung (14) generiert und wobei der Projektgenerator (12) unter Verwendung der Entwicklungsumgebung (14) anhand von vorgebbaren oder vorgegebenen zweiten Daten, wie insbesondere Planungsdaten (17), Erfahrungsdaten (20) und Hardwarewissen (21), eine Steuerungssoftware (15) generiert.
8. Verfahren nach Anspruch 7, wobei der Generator (10) weiter einen Anlagenkonfigurator (13) umfasst und wobei der Anlagenkonfigurator (13) anhand von vorgebbaren oder vorgegebenen dritten Daten, wie insbesondere Hardwarewissen (21), eine Konfiguration

(16) einer Automatisierungshardware ermittelt und ggf. vornimmt.

9. Verfahren nach Anspruch 7 oder 8, wobei die ersten Daten Planungsdaten (17), Programmwissen (18) und/oder Verwaltungswissen (19) umfassen.

5

10. Verfahren nach Anspruch 7 oder 8, wobei die zweiten Daten Planungsdaten (17), Erfahrungsdaten (20) und/oder Hardwarewissen (21) umfassen.

11. Verfahren nach Anspruch 7 oder 8, wobei die dritten Daten Hardwarewissen (21) umfassen.

10

12. Verfahren nach einem der Ansprüche 7 bis 11, wobei die Steuerungssoftware (16) eine Mensch-Maschine-Schnittstelle umfasst.

---

Hierzu 3 Seite(n) Zeichnungen

---

15

20

25

30

35

40

45

50

55

60

65

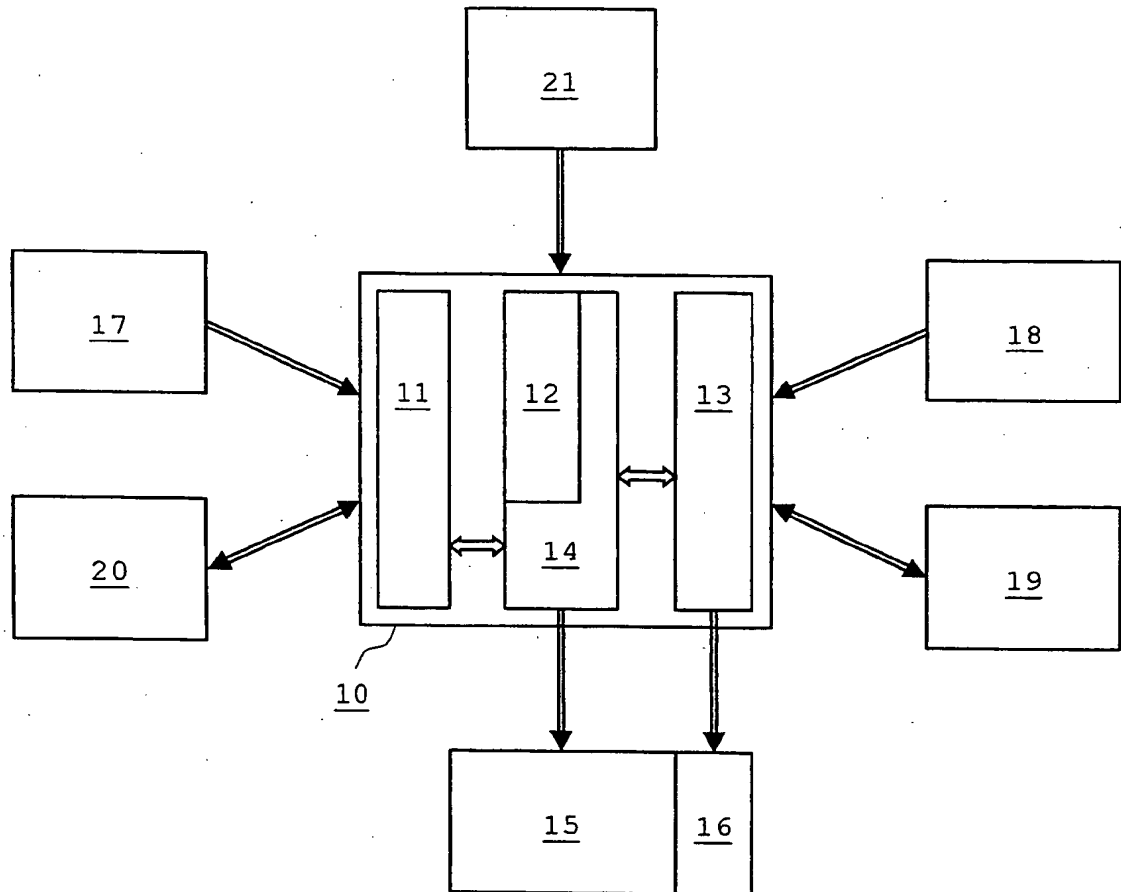


FIG 1



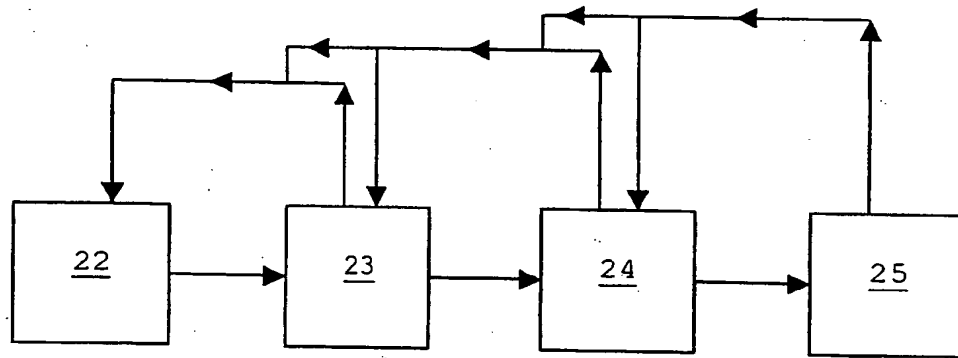


FIG 2

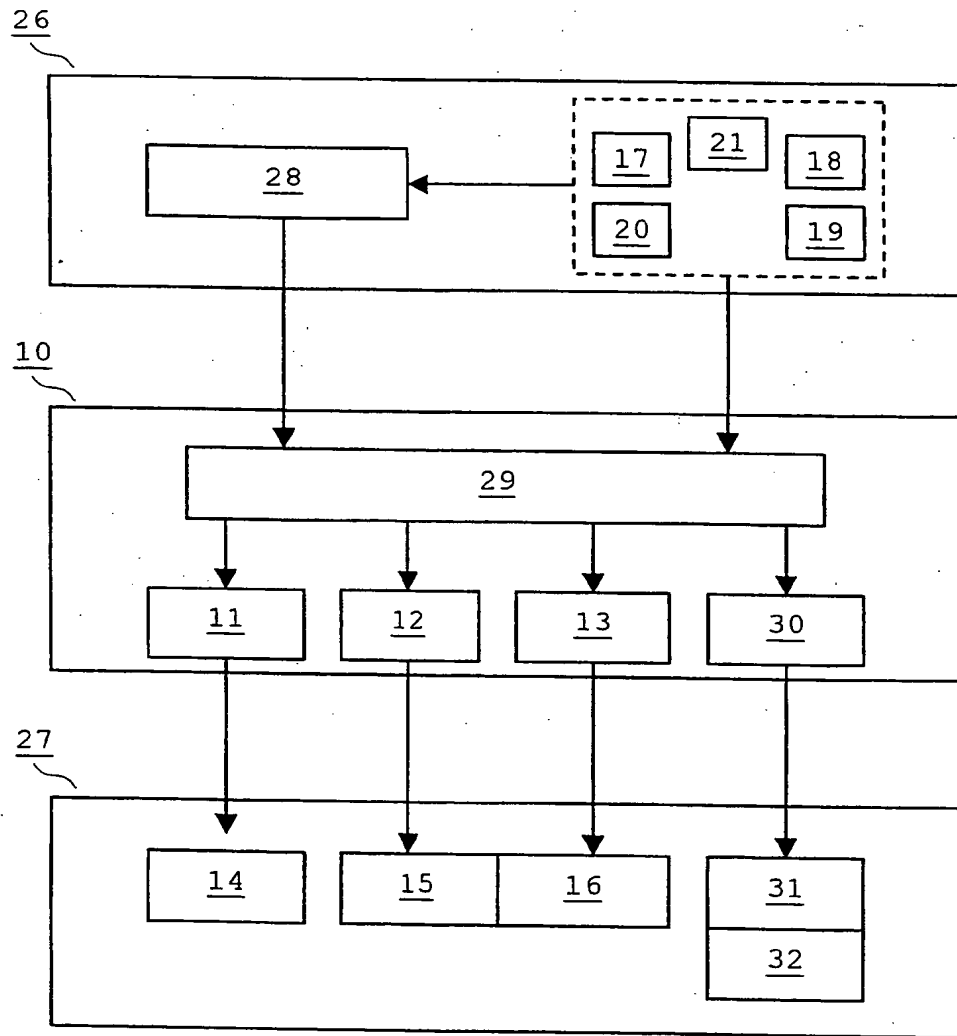


FIG 3